ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ
INFORMATION TECHNOLOGY, COMPUTER SCIENCE AND MANAGEMENT



Check for updates

УДК 004.896
https://doi.org/10.23947/2687-1653-2023-23-2-169-179

Научная статья



Прогнозирование поведения участников дорожного движения в условиях проселочных дорог для беспилотных автомобилей

С.А. Иванов ✉, **Б. Рашид**
Центр беспилотных технологий университета Иннополис, г. Иннополис, Российская Федерация
✉ se.ivanov@innopolis.ru

Аннотация
Введение. Благодаря модулю прогнозирования траекторий движения динамических объектов беспилотный автомобиль способен безопасно двигаться по дорогам общего пользования. Однако все современные методы прогнозирования оценивают производительность только в городских условиях и не рассматривают свою применимость к домену проселочных дорог. Цель данного исследования заключается в анализе адаптивности существующих методов прогнозирования и разработке подхода, который будет демонстрировать лучшую производительность при работе в новых условиях.
Материалы и методы. В качестве решения предлагается использовать нейронную сеть, включающую в себя следующие подмодули: графовый кодировщик сцены, мультимодальный декодировщик траекторий, модуль фильтрации траекторий. Также предлагается применить адаптированную функцию потерь, которая штрафует сеть за генерацию траекторий, выходящих за границы дорожного полотна. Данные элементы задействуют распространённые практики решения задачи прогнозирования, а также адаптируют её для домена проселочных дорог.
Результаты исследования. Проанализированы основные отличия и условия работы модуля прогнозирования в условиях проселочных дорог. Выполнена симуляция нового домена путем модификации существующих наборов данных. Проведено сравнение популярных методов прогнозирования и оценена их применимость к новым условиям. Представлен новый, более адаптивный к новому домену, подход.
Обсуждение и заключение. Проведенное сравнение с другими популярными методами показывает, что предложенное авторами решение обеспечивает более точные результаты прогнозирования. Также были выявлены недостатки предложенного подхода и описаны возможные пути их устранения.

Ключевые слова: прогнозирование траекторий, прогнозирование поведения, нейронные сети, беспилотные автомобили, искусственный интеллект, автономные автомобили

Благодарности: авторы выражают благодарность центру беспилотных технологий университета Иннополис за помощь в проведении исследования.

Для цитирования. Иванов С.А., Рашид Б. Прогнозирование поведения участников дорожного движения в условиях проселочных дорог для беспилотных автомобилей. *Advanced Engineering Research (Rostov-on-Don)*. 2023;23(2):169–179. https://doi.org/10.23947/2687-1653-2023-23-2-169-179

Original article

Predicting the Behavior of Road Users in Rural Areas for Self-Driving Cars

Sergey A. Ivanov ✉, **Bader Rasheed**
Laboratory of Unmanned Technology, Innopolis University, Innopolis, Russian Federation
✉ se.ivanov@innopolis.ru

Abstract
Introduction. The prediction module generates possible future trajectories of dynamic objects that enables a self-driving vehicle to move safely on public roads. However, all modern prediction methods evaluate their performance only under urban conditions and do not consider their applicability to the domain of rural roads. This work examined the adaptability of existing methods to work under rural unstructured conditions and suggested a new, improved approach.
Materials and Methods. As a solution, we propose to use a neural network that includes the following submodules: a graph-based scene encoder, a multimodal trajectory decoder, and a trajectory filtering module. Another proposed feature is to use an adapted loss function that penalizes the network for generating trajectories that go beyond the drivable area. These elements use standard practices for solving the prediction problem and adapting it to the domain of rural roads.
Results. The presented analysis described the basic features of the prediction module in the rural road domain, showed a comparison of popular models, and discussed its applicability to new conditions. The paper describes the new approach that is more adaptive to the considered domain of study. A simulation of the new domain was performed by modifying existing public datasets.
Discussion and Conclusion. Comparison to other popular methods has shown that the proposed approach provides more accurate results. The disadvantages of the proposed approach were also identified and possible solutions were described.

Keywords: trajectory prediction, behavior prediction, neural networks, self-driving cars, artificial intelligence, autonomous cars

Acknowledgements: the authors appreciate the "Center for Autonomous Technologies", Innopolis University, for their assistance in conducting the research.

For citation. Ivanov SA, Rasheed B. Predicting the Behavior of Road Users in Rural Areas for Self-Driving Cars. *Advanced Engineering Research (Rostov-on-Don).* 2023;23(2):169–179. https://doi.org/10.23947/2687-1653-2023-23-2-169-179

Введение. Последние достижения в области искусственного интеллекта (ИИ) активно внедряются в различные сферы деятельности. Одно из таких достижений — беспилотные транспортные средства (БТС). Целью текущих исследований является создание алгоритмов, позволяющих БТС безопасно двигаться по дорогам общего пользования. Это позволит значительно снизить количество дорожно-транспортных происшествий [1].

Научное сообщество уже выделило основные модули беспилотного автомобиля. Один из них — система прогнозирования будущего поведения участников дорожного движения (агентов) [2]. Четкое понимание того, как будет развиваться окружение и в какую сторону будут двигаться динамические объекты (пешеходы, автомобили, велосипедисты), крайне необходимо БТС для поиска и использования безопасной и эффективной траектории движения.

Множество научных статей посвящено задаче прогнозирования таких траекторий [3–12]. Однако в настоящее время не проводится активных исследований по применению существующих методов вне городских условий. А делать это крайне важно, поскольку автономные автомобили будут использоваться и на проселочных дорогах тоже [13]. Городские условия сильно структурированы: автомобили в основном следуют по полосам движения, а пешеходы движутся по специальным зонам. В этом смысле область проселочных дорог является полной противоположностью, а значит, она будет иметь дополнительные сложности при разработке. В данной работе сосредоточено внимание именно на этих сложностях: рассматриваются существующие методы прогнозирования и их применимость к новым, менее структурированным условиям.

Цель исследования заключается в следующем:

– анализ основных отличий и условий работы модуля прогнозирования в условиях проселочных дорог;

– симуляция менее структурированного домена проселочных дорог путем модификации существующих наборов данных;

– сравнение современных методов прогнозирования, в том числе на предмет их применимости к новым условиям;

– описание нового подхода и доказательство его более высокой точности работы в сравнении с другими методами прогнозирования.

Материалы и методы. На первый взгляд, представленный домен может показаться более простой версией городских условий из-за того, что проселочные дороги характеризуются меньшим дорожным трафиком. Однако отсутствие сложных дорожных развязок, специальных пешеходных зон, большого количества знаков, разметки и т. д. делает домен проселочных дорог менее структурированным, то есть меньшее количество правил и специфических шаблонов движения увеличивает хаотичность и снижает предсказуемость поведения машин и пешеходов.

Следующие особенности домена проселочных дорог будут оказывать сильное влияние на выбор архитектуры модуля прогнозирования:

– наличие перекрестков. Несомненно, они на проселочных дорогах более простые в сравнении с городскими, но в то же время этот факт простоты означает, что модель должна учитывать мультимодальность и оценивать вероятность выбора каждого возможного направления движения на перекрестке при подъезде агента к нему;

– проселочные дороги не имеют разметки полос движения, пешеходных переходов, велосипедных дорожек и т. д. Вместо нее HD-карта будет содержать только информацию о границах дорожного полотна. Следовательно, этап кодирования сцены должен учитывать эту особенность, чтобы эффективнее описывать окружающий контекст;

– пешеходы и велосипедисты будут двигаться по одной дороге с обычными и беспилотными автомобилями. Следовательно, модель должна быть адаптивной для предсказания будущих траекторий движения как автомобилей, так и пешеходов/велосипедистов.

Модуль прогнозирования подразумевает наличие систем распознавания, трекинга и локализации БТС и их точную работу. Авторы статьи задействуют датасет Argoverse, который хранит необходимые записи работы всех систем в удобном виде [14].

Датасет состоит из записей дорожных сцен, зафиксированных на улицах Майами и Питтсбурга США). Каждая из записей содержит локальную часть карты местности (границы полос, дорог, пешеходные переходы) и список всех распознанных агентов, включающий текущую позицию и историю передвижений каждого из них. Каждая из записей делится на две части: две секунды истории наблюдений и три последующих секунды, для которых делается прогноз (горизонт прогнозирования). Данные будущего движения объектов также доступны и используются для вычисления точности методов прогнозирования и тренировки моделей.

Информация об агентах представлена в дискретном формате. Временной интервал между измерениями фиксирован, в этой работе он равен 0,1 секунды (10 Гц).

Для каждого момента времени t модуль получает историю наблюдений S_i^p для каждого обнаруженного агента i. История наблюдений состоит из текущего и прошлых состояний агента, где каждое из состояний s_t — это 2D позиция в глобальной системе координат. Авторы делают допущение, что информация о высоте является избыточной.

Датасет также предоставляет доступ к HD-карте, которая содержит информацию о границах полос и дорожном полотне, пешеходных переходах. Для симуляции домена проселочных дорог выполнена модификация датасета таким образом, чтобы исключить из дорожных карт всю информацию, кроме границ дорожного полотна D. Тем самым уменьшено количество информации о дорожном контексте и усложнена задача прогнозирования.

Следовательно, контекст сцены представляется как:

$$c = \left\{ S_1^p, S_2^p, \ldots, S_k^p, D \right\}, \tag{1}$$

где k — это общее количество отслеживаемых агентов на сцене.

Такой подход подразумевает прогнозирование траектории только для одного агента за одно выполнение, поэтому далее S_i^p трактуется как S^p для упрощения. Чтобы обобщить модель для всех распознанных агентов, требуется повторить предложенный подход для всех k агентов на сцене. Агент, для которого в текущий момент делается прогноз, считается целевым агентом.

Для оценки точности работы методов прогнозирования датасет содержит записанные будущие траектории S^f для каждого целевого агента:

$$S^f = \left\{ s_1, s_2, \ldots, s_H \right\}, \tag{2}$$

где H обозначает количество следующих временных шагов. В данном случае параметр H будет равен 30, так как горизонт планирования равен трем секундам с частотой дискретизации 10 Гц.

Домен работы модуля прогнозирования мультимодален, то есть в абсолютно идентичных дорожных ситуациях будущее поведение агентов может существенно различаться. Например, автомобиль, приближающийся к перекрестку, может продолжить свое движение прямо или выполнить поворот. Чтобы принять это во внимание, на выходе модели необходимо генерировать M возможных будущих траекторий и M вероятностей выполнения каждой из них.

Следовательно, цель модуля прогнозирования — создать такую функцию f, которая принимает на вход контекст сцены c и генерирует M пар возможных будущих траекторий и их вероятностей:

$$f(c)=\left\{S_1^f, S_2^f, \ldots, S_M^f, p_1, p_2, \ldots, p_M\right\} \tag{3}$$

При этом хотя бы одна генерируемая траектория S^f должна быть максимально приближена к реальной траектории S^f, а вероятность её выполнения p должна быть приближена к единице.

Архитектура модели. Предлагаемый подход подразумевает использование нейронной сети, состоящей из подмодулей кодирования сцены, декодирования и фильтрации траекторий. Архитектура системы представлена на рис. 1.

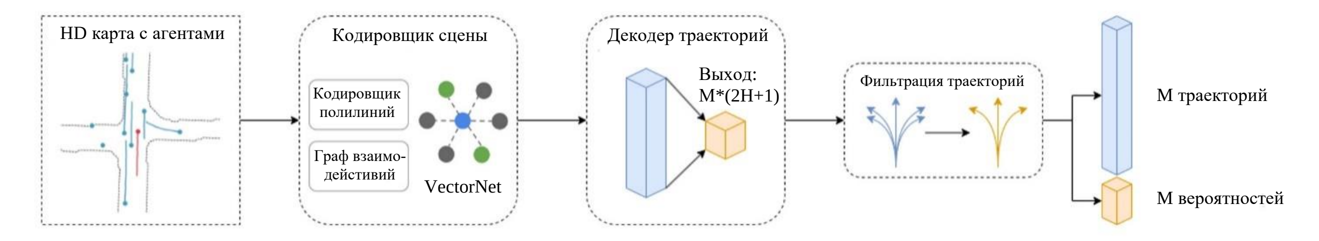


Рис. 1. Архитектура системы

За кодирование сцены отвечает адаптированная к новым условиям нейронная сеть, основанная на векторном представлении. Данный выбор обусловлен тем, что на проселочных дорогах HD-карта будет содержать ограниченное количество информации (только границы дорожного полотна и историю наблюдений динамических объектов). Популярные методы представляют контекст дорожной сцены c в формате изображения и обрабатывают его с помощью сверточных нейронных сетей, однако векторное кодирование позволяет избежать накладных расходов, которые связанны с генерацией изображения [4–5].

Представленный кодировщик основан на модели VectorNet, однако его входной формат данных был модифицирован, чтобы получать информацию только о границах дорожного полотна и состояниях агентов [3]. Данный кодировщик представляет границы дороги и состояния агентов с помощью ломанных линий, которые далее обрабатываются графовой нейронной сетью. Это позволяет кодировать взаимодействие между ломаными линиями. Подробности реализации описаны в статье VectorNet [3].

Декодировщик траекторий представляет собой задачу регрессии нескольких возможных траекторий и генерирования набора вероятностей. Для решения данной задачи задействована модель многослойного персептрона. Реализация декодировщика вдохновлена моделью MTP [4], однако авторы статьи предлагают другую формулу вычисления наилучшей траектории m^* из множества M траекторий. Также предлагается использовать дополнительный механизм, который штрафует модель за прогнозы, выходящие за пределы области движения.

Авторы оригинальной модели MTP предлагают тренировать многослойный персептрон с помощью функции потерь, которая представляет сумму $\mathcal{L}_{reg}$ и $\mathcal{L}_{class}$, где:

$$\mathcal{L}_{reg}=\mathcal{L}\left(S^f, S_{m^*}^f\right) \tag{4}$$

$$\mathcal{L}_{class}=-\sum_{m=1}^{M} I_{m=m^*} \log p_m \tag{5}$$

В данном случае $\mathcal{L}_{reg}$ — это среднеквадратичная ошибка между реальной траекторией S^f и лучшей траекторий m^* из M генерируемых.

$$\mathcal{L}\left(S^f, S_{m^*}^f\right)=\frac{1}{H}\sum_{i=1}^{H}\left(s_i-s_i^*\right)^2, \tag{6}$$

где s_i — реальная будущая позиция агента в момент времени i, а $\tilde{s}_i^*$ — прогнозируемое будущее состояние наилучшей траектории m^*.

$\mathcal{L}_{class}$ — функция потерь, основанная на перекрёстной энтропии, которая увеличивает вероятность выполнения лучшей из прогнозируемых траекторий m^* до 1 и уменьшает вероятность других траекторий до 0.

I_c является бинарным индикатором, равным 1, если условие c верно, и 0 — в противном случае.

В оригинальной статье лучшая из прогнозируемых траекторий m^* определяется как та, которая имеет минимальное значение среднеквадратичной ошибки в сравнении с реальной траекторией:

$$m = \underset{m}{argmin}\ \mathcal{L}\left(S^f, S_m^f\right). \tag{7}$$

Авторы статьи предлагают использовать следующую модификацию:

$$m = \underset{m\in\Delta}{argmin}\ \mathcal{L}\left(S^f, S_m^f\right), \tag{8}$$

где Δ — подмножество генерируемых траекторий, которое имеет схожее финальное направление с реальной траекторией S^f.

Идея заключается в том, чтобы при расчете наилучшей траектории m^* убрать из рассмотрения траектории, в которых финальное направление агента значительно отличается от направления в реальной траектории. Если разница направлений отличается меньше, чем некоторый порог γ, то генерируемая траектория считается корректной, то есть $m \subset \Delta$. В рассматриваемом случае $\gamma = 30^o$. Следовательно, лучшая траектория m^* должна иметь схожее финальное направление и наименьшее значение функции потерь.

В данной работе также задействованы предварительные знания о домене (prior knowledge) для достижения большей сходимости модели [15]. Поскольку при движении в домене проселочных дорог с HD-карты доступна только информация о границах дорожного полотна, вводится дополнительная переменная $\mathcal{L}_{da}$ в функцию потерь. Благодаря ей модель будет штрафовать прогнозируемые траектории, которые выходят за пределы дороги в случаях, если хотя бы одно состояние $s_i \notin D$. Модель штрафует только наилучшую траекторию, поскольку только для неё можно определить направление уменьшения ошибки путем приближения наилучшей из генерируемых траекторий m^* к реальной траектории S^f.

Следовательно, $\mathcal{L}_{da}$ определяется как:

$$\mathcal{L}_{da} = \frac{1}{H}\sum_{i=1}^{H} I_c \cdot \left(s_i - s_i^*\right)^2, \tag{9}$$

где I_c равен 1, если $s_i \notin D$, и 0 — в обратном случае.

Финальная функция потерь определяется как:

$$\mathcal{L} = \mathcal{L}_{class} + \alpha \cdot \mathcal{L}_{reg} + \beta \cdot \mathcal{L}_{da}, \tag{10}$$

где α и β — гиперпараметры нейронной сети, использующиеся для обучения. В данном случае оба этих параметра равны 0,5.

Для фильтрации похожих и дублирующихся траекторий предлагаемый подход использует фильтрацию конечного набора траекторий M на финальном этапе. Этот модуль необходим, поскольку в некоторых случаях количество возможных траекторий агента может быть меньше M, например, когда автомобиль движется по прямой дороге с постоянной скоростью, модель может сгенерировать только одну траекторию: автомобиль продолжает двигаться прямо. Однако необходимость генерировать именно M траекторий приведет к тому, что все прогнозы будут схожи между собой.

Предлагаемая фильтрация основана на финальном направлении и позициях состояний s_i: если направление и сумма отклонений между состояниями s_i реальной и генерируемой траекторий меньше порогового значения σ, то траектории считаются схожими. Авторы усредняют каждое состояние траекторий и суммируют вероятности траекторий p_i.

Такой подход был реализован на языке программирования Python на фреймворке глубокого обучения Pytorch. Модель обучалась на видеокарте GeForce RTX 2080 Ti в течение 40 эпох, тренировка заняла четыре часа.

Результаты исследования. Для оценки точности моделей прогнозирования в этом разделе используются широко распространённые метрики для задачи прогнозирования траекторий: среднее значение отклонения (average displacement error, ADE), финальное отклонение (final displacement error, FDE) [6], MissRate (MR) и Offroad rate (OR).

Для мультимодальных случаев с генерацией нескольких траекторий ADE и FDE принимается как минимальное ADE и FDE среди M траекторий (траектория с наименьшим значением метрик) [5].

Метрика MissRate рассчитывается как процент «промазанных» прогнозов. Прогноз считается «промазанным», если метрика ADE сгенерированной траектории более двух метров. Метрика OR рассчитывается как процент траекторий, в которой хотя бы одно состояние s_i выходит за пределы области движений D.

Для визуализации контекста сцены c, а также реальной и прогнозируемых будущих траекторий S^f и S^f был реализован скрипт на языке программирования Python с использованием библиотеки Matplotlib.

В этом разделе сравнивается работа нескольких разных методов в случае неструктурированного домена. В сравнении задействованы следующие методы:

– фильтр Калмана;

– предложенный кодировщик сцены с генерацией одной единственной траектории (Single trajectory output);

– предложенный кодировщик сцены со сведением задачи к классификации среди предопределенных траекторий: по наборам из 64 и 415 предопределенных траекторий (Fixed set classification);

– предлагаемый подход (Proposed approach).

В таблице 1 представлено сравнение точности методов при работе в неструктурированных условиях. Сравнению подвергается несколько методов, включая предлагаемый подход.

Таблица 1

Сравнение моделей в неструктурированном домене работы

Method	Modes	ADE_1	FDE_1	ADE_6	FDE_6	$MR2_1$	$MR2_6$	OR
Kalman filter	1	3,78	8,05	3,78	8,05	0,89	0,89	5,89
Single trajectory output	1	3,12	6,75	3,12	6,75	0,89	0,89	3,26
Fixed set classification	415	3,27	7,00	1,74	3,57	0,84	0,52	3,61
Fixed set classification	64	2,6	5,63	1,52	2,91	0,82	0,49	2,58
Proposed approach	6	2,36	5,29	1,32	2,55	0,78	0,38	1,84

Фильтр Калмана. Простейший способ прогнозирования поведения заключается в получении текущего состояния объекта (текущая полоса движения, скорость, направление и т. д.) и распространении этого состояния на будущие шаги на основе некоторых допущений, например, что автомобиль будет продолжать следовать по своей полосе или будет иметь постоянную скорость и/или ускорение. Другим популярным методом для таких задач является использование фильтра Калмана [12].

Согласно данным таблицы 1, фильтр Калмана работает хуже, чем все представленные методы, основанные на нейронных сетях.

На рис. 2 представлены два случая. В первом случае фильтр Калмана успешно выполняет прогнозирование, поскольку транспортное средство движется прямо, без каких-либо поворотов или изменений скорости. Во втором случае фильтр Калмана приводит ошибочный прогноз из-за отсутствия знаний о контексте дорожной ситуации.

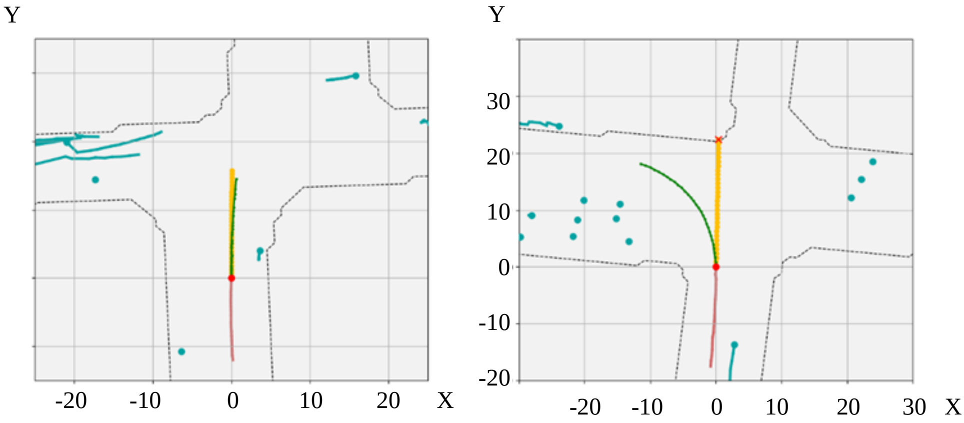


Рис. 2. Пример предсказаний с помощью фильтра Калмана. Пунктирные линии — граница дорожного полотна, линии красного цвета — целевой агент с историей наблюдений, голубого — другие агенты, зеленого — реальная траектория, желтого — прогнозируемая траектория, красные кресты указывают предсказанные состояния за пределами дорожного полотна

Single trajectory output. Данный метод подразумевает использование графового кодировщика сцены, который идентичен используемому в предложенном подходе. Вывод сети подразумевает генерацию только одной траектории. Данная модель тренируется с использованием среднеквадратичной функции потерь.

Как показано в таблице 1, нейронная сеть даже с генерированием единственной траектории демонстрирует лучшие результаты в сравнении с фильтром Калмана.

На рис. 3 представлена визуализация работы данного метода прогнозирования. Изображение слева показывает, что модель может успешно предсказывать поворот агента. Изображение справа показывает, что генерации одной траектории недостаточно. Нейронная сеть пытается представить оба возможных исхода: движение прямо и поворот направо. В результате модель выводит среднее значение двух исходов.

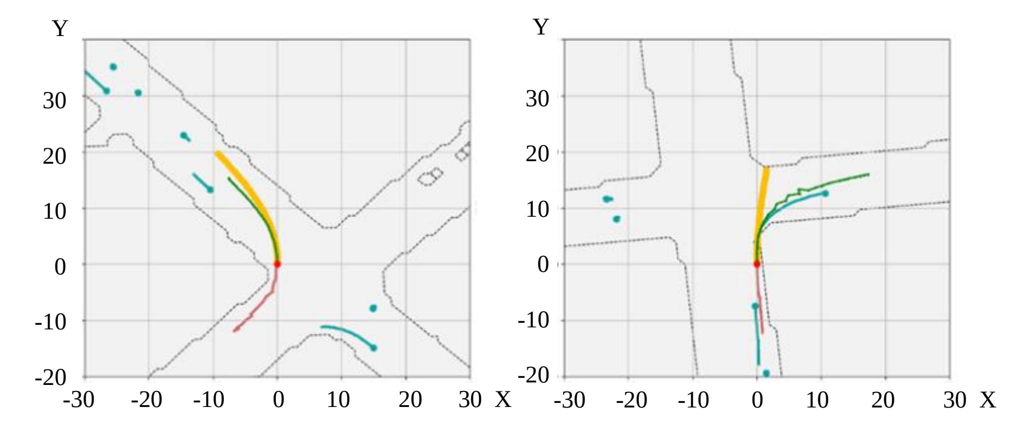


Рис. 3. Пример генерации одной траектории. Линией красного цвета показан целевой агент с историей наблюдений, зеленого — реальная траектория движения, желтого — прогнозируемая траектория

Fixed set classification. Реализация была вдохновлена методом прогнозирования CoverNet [5]. Данная модель состоит из предлагаемого векторного кодировщика сцены, за которым следует иной декодер траекторий. Декодер представляет собой задачу классификации по предопределенному набору траекторий, состоящему из физически реализуемых траекторий транспортного средства с достаточным покрытием. Для экспериментов были созданы два набора: из 415 и 64 возможных траекторий. Второй набор имеет такое же покрытие, как и первый, но обеспечивает меньшую плотность траекторий. Подробная информация о наборах траекторий содержится в статье CoverNet [5].

Визуализация работы представлена на рис. 4. Модель классификации успешно справляется с мультимодальностью на перекрестках, но в некоторых случаях отсутствие достаточного покрытия набором траекторий негативно влияет на результаты.

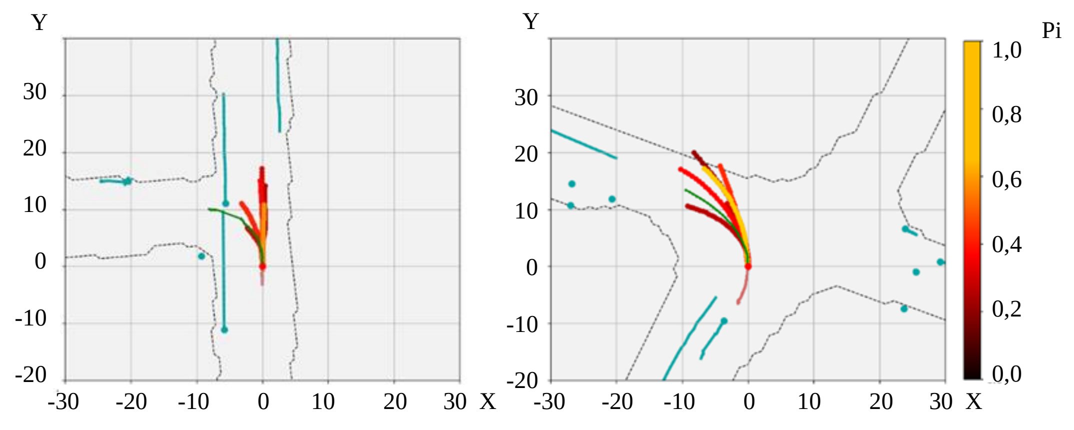


Рис. 4. Пример прогнозирования с помощью классификационной модели. Линии красного цвета — целевой агент с историей наблюдений, зеленого — реальная траектория. *M* прогнозируемых траекторий с разной вероятностью выполнения p_i представлены с использованием красно-желтых оттенков

Как показано в таблице 1, данный метод работает точнее, чем генерация одной траектории, но хуже, чем предлагаемый подход. Кроме того, увеличение плотности набора траекторий за счет использования набора из 415 траекторий не улучшило результаты. Авторы связывают это с наличием шума в наборе данных, который исходит от системы отслеживания, используемой при сборе данных.

Proposed approach. Предлагаемый подход устраняет недостатки всех вышеописанных методов. Это мультимодальный способ прогнозирования, который не страдает от ограничений предопределенного набора траекторий.

Кроме того, согласно таблице 1, предлагаемый подход превосходит все остальные методы по всем показателям. Как показано на рис. 5, метод успешно захватывает два возможных исхода на перекрестке: движение прямо или выполнение поворота.

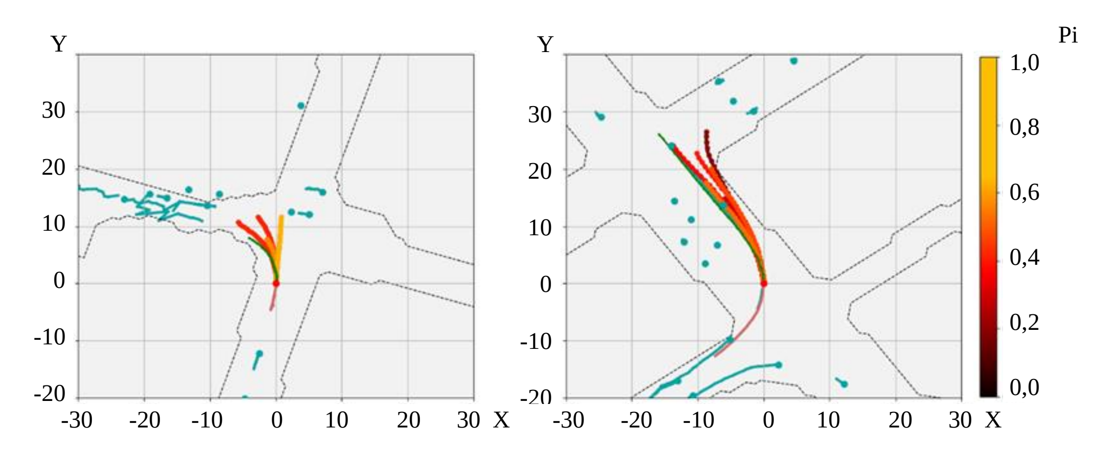


Рис. 5. Пример прогнозирования с помощью классификационной модели. Линии красного цвета — целевой агент с историей наблюдений, зеленого — реальная траектория. *M* прогнозируемых траекторий с разной вероятностью выполнения p_i представлены с использованием красно-желтых оттенков

На рис. 6 показан пример фильтрации схожих траекторий в случае единственного возможного исхода. Вероятность того, что агент завершит начатый поворот, близка к единице, поскольку он уже находится в процессе поворота. Следовательно, в этом случае вероятность выполнения других исходов близка к 0. Предлагаемый модуль успешно фильтрует схожие траектории.

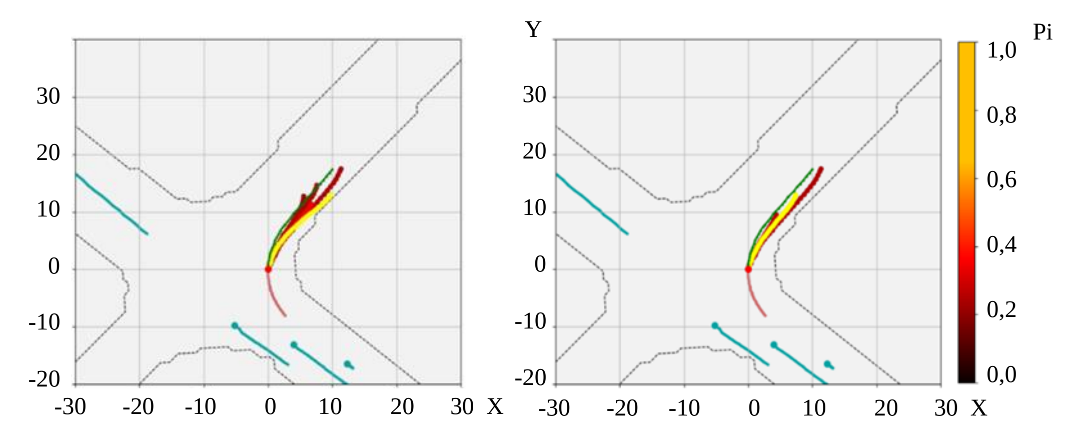


Рис. 6. Эффект фильтрации. Слева представлен весь набор прогнозов, справа — только отфильтрованный набор

Ограничения. Хотя авторы оригинальной статьи о модели MTP [4] указывают, что их метод решает проблему коллапса режима (mode collapse), проведенные авторами данной статьи эксперименты этого не подтверждают. Такая проблема все еще возникает в некоторых случаях. Предполагается, что это связанно со следующими особенностями: функция потерь не штрафует нейронную сеть за генерацию всех возможных траекторий, которые может исполнить целевой агент, пока лучшая из них максимально близко к реальной траектории. Но также модель никак не поощряет сеть к прогнозированию разнообразных возможных путей. Следовательно, сети выгодно делать несколько похожих прогнозов в одном направлении, в котором она более уверена, чем делать по одному прогнозу для каждого возможного пути.

Одним из возможных решений данной проблемы может быть использование декодера траекторий, представленного в моделях TnT, DenseTnT [10–11], которые подразумевают генерацию финальных целей на первых этапах работы. В данных моделях сначала генерируются все возможные финальные цели для агента, а далее генерируются траектории, которые описывают движение от стартовой позиции до каждой из целей. Это позволяет отфильтровывать похожие финальные цели на ранних стадиях и предотвращать коллапс режима.

Обсуждение и заключение. В выполненной работе исследованы современные методы решения задачи прогнозирования траектории. Рассмотрена адаптивность методов к неструктурированным дорожным

условиям — проселочным дорогам. Выявлена недостаточная точность методов, и предложен новый подход к прогнозированию.

Предлагаемый подход основан на моделях VectorNet и MTP, но был адаптирован для домена проселочных дорог. Кроме того, был предложен модуль фильтрации траекторий и дополнительный механизм для функции потерь, который штрафует траектории за выход за пределы зоны движения.

Представленное сравнение показывает, что предлагаемый подход превосходит другие популярные методы.

Были выявлены ограничения подхода MTP: выходные данные по-прежнему имеют тенденцию к коллапсу режима. Предложение для дальнейших модификаций заключается в использовании методов, которые генерируют финальную цель на ранних этапах прогнозирования и тем самым меньше подвержены коллапсу режима.

Список литературы

1. Qing Rao, Jelena Frtunikj. *Deep Learning for Self-Driving Cars: Chances and Challenges*. In: Proc. 1st International Workshop on Software Engineering for AI in Autonomous Systems. New York, NY: Association for Computing Machinery; 2018. P. 35–38. https://doi.org/10.1145/3194085.3194087

2. Shaoshan Liu, Liyun Li, Jie Tang, et al. *Creating Autonomous Vehicle Systems*. San Rafael, CA: Morgan & Claypool; 2020. 216 p.

3. Jiyang Gao, Chen Sun, Hang Zhao, et al. *VectorNet: Encoding HD Maps and Agent Dynamics from Vectorized Representation.* In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 11525–11533. https://doi.org/10.48550/arXiv.2005.04259

4. Henggang Cui, Vladan Radosavljevic, Fang-Chieh Chou, et al. *Multimodal Trajectory Predictions for Autonomous Driving Using Deep Convolutional Networks*. In: Proc. IEEE International Conference on Robotics and Automation (ICRA). Montreal, BC: IEEE; 2019. P. 2090–2096. https://doi.org/10.48550/arXiv.1809.10732

5. Tung Phan-Minh, Elena Corina Grigore, Freddy A. Boulton, et al. *CoverNet: Multimodal Behavior Prediction Using Trajectory Sets*. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 14074–14083. https://doi.org/10.48550/arXiv.1911.10298

6. Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, et al. *Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction*. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 14424–14432. https://doi.org/10.48550/arXiv.2002.11927

7. Biktairov Yu., Stebelev M., Rudenko I., et al. *PRANK: Motion Prediction Based on RANKing*. In: Neural Information Processing Systems. Vancouver: Virtual Conference; 2020. P. 2553–2563. https://doi.org/10.48550/arXiv.2010.12007

8. Yuning Chai, Benjamin Sapp, Mayank Bansal, et al. MultiPath: Multiple Probabilistic Anchor Trajectory Hypotheses for Behavior Prediction. *Proceedings of the Conference on Robot Learning*. 2020;100:86–99. https://doi.org/10.48550/arXiv.1910.05449

9. Ajay Jain, Sergio Casas, Renjie Liao, et al. Discrete Residual Flow for Probabilistic Pedestrian Behavior Prediction. In: Proc. 3rd Conference on Robot Learning. *Proceedings of Machine Learning Research*. 2019;100:407–419. https://doi.org/10.48550/arXiv.1910.08041

10. Hang Zhao, Jiyang Gao, Tian Lan, et al. *TNT: Target-driveN Trajectory Prediction*. In: Conference on Robot Learning. Cambridge, MA: Virtual Conference; 2020. P. 895–904. https://doi.org/10.48550/arXiv.2008.08294

11. Junru Gu, Chen Sun, Hang Zhao. *Dense TNT: End-to-end Trajectory Prediction from Dense Goal Sets*. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal, BC: IEEE; 2021. P. 15303–15312. https://doi.org/10.48550/arXiv.2108.09640

12. Prévost C.G., Desbiens A., Gagnon E. *Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle*. In: Proceedings of the American Control Conference. New York, NY: IEEE; 2007. P. 1805–1810. https://doi.org/10.1109/ACC.2007.4282823

13. Zeyu Zhu, Nan Li, Ruoyu Sun, et al. *Off-road Autonomous Vehicles Traversability Analysis and Trajectory Planning Based on Deep Inverse Reinforcement Learning*. In: IEEE Intelligent Vehicles Symposium (IV). Las Vegas, NV: IEEE; 2020. P. 971–977. https://doi.org/10.1109/IV47402.2020.9304721

14. Mig-Fang Chang, John Lambert, Patsorn Sangkloy, et al. *Argoverse: 3D Tracking and Forecasting with Rich Maps*. In: Proc. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA: IEEE; 2019. P. 8748–8757. https://doi.org/10.1109/CVPR.2019.00895

15. Casas S., Gulino C., Suo S., et al. *The Importance of Prior Knowledge in Precise Multimodal Prediction*. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, NV: IEEE; 2020. P. 2295–2302. https://doi.org/10.48550/arXiv.2006.02636

References

1. Qing Rao, Jelena Frtunikj. *Deep Learning for Self-Driving Cars: Chances and Challenges*. In: Proc. 1st International Workshop on Software Engineering for AI in Autonomous Systems. New York, NY: Association for Computing Machinery; 2018. P. 35–38. https://doi.org/10.1145/3194085.3194087

2. Shaoshan Liu, Liyun Li, Jie Tang, et al. *Creating Autonomous Vehicle Systems*. San Rafael, CA: Morgan & Claypool; 2020. 216 p.

3. Jiyang Gao, Chen Sun, Hang Zhao, et al. *VectorNet: Encoding HD Maps and Agent Dynamics from Vectorized Representation.* In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 11525–11533. https://doi.org/10.48550/arXiv.2005.04259

4. Henggang Cui, Vladan Radosavljevic, Fang-Chieh Chou, et al. *Multimodal Trajectory Predictions for Autonomous Driving Using Deep Convolutional Networks*. In: Proc. IEEE International Conference on Robotics and Automation (ICRA). Montreal, BC: IEEE; 2019. P. 2090–2096. https://doi.org/10.48550/arXiv.1809.10732

5. Tung Phan-Minh, Elena Corina Grigore, Freddy A. Boulton, et al. *CoverNet: Multimodal Behavior Prediction Using Trajectory Sets*. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 14074-14083. https://doi.org/10.48550/arXiv.1911.10298

6. Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, et al. *Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction*. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE; 2020. P. 14424–14432. https://doi.org/10.48550/arXiv.2002.11927

7. Biktairov Yu, Stebelev M, Rudenko I, et al. *PRANK: Motion Prediction Based on RANKing*. In: Neural Information Processing Systems. Vancouver: Virtual Conference; 2020. P. 2553–2563. https://doi.org/10.48550/arXiv.2010.12007

8. Yuning Chai, Benjamin Sapp, Mayank Bansal, et al. MultiPath: Multiple Probabilistic Anchor Trajectory Hypotheses for Behavior Prediction. *Proceedings of the Conference on Robot Learning*. 2020;100:86–99. https://doi.org/10.48550/arXiv.1910.05449

9. Ajay Jain, Sergio Casas, Renjie Liao, et al. Discrete Residual Flow for Probabilistic Pedestrian Behavior Prediction. In: Proc. 3rd Conference on Robot Learning, Osaka, Japan, 2019. *Proceedings of Machine Learning Research*. 2019;100:407–419. https://doi.org/10.48550/arXiv.1910.08041

10. Hang Zhao, Jiyang Gao, Tian Lan, et al. *TNT: Target-driveN Trajectory Prediction*. In: Conference on Robot Learning. Cambridge, MA: Virtual Conference; 2020. P. 895–904. https://doi.org/10.48550/arXiv.2008.08294

11. Junru Gu, Chen Sun, Hang Zhao. *Dense TNT: End-to-end Trajectory Prediction from Dense Goal Sets*. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal, BC: IEEE; 2021. P. 15303–15312. https://doi.org/10.48550/arXiv.2108.09640

12. Prévost CG, Desbiens A, Gagnon E. *Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle*. In: Proc. American Control Conference. New York, NY: IEEE; 2007. P. 1805–1810. https://doi.org/10.1109/ACC.2007.4282823

13. Zeyu Zhu, Nan Li, Ruoyu Sun, et al. *Off-road Autonomous Vehicles Traversability Analysis and Trajectory Planning Based on Deep Inverse Reinforcement Learning*. In: IEEE Intelligent Vehicles Symposium (IV). Las Vegas, NV: IEEE; 2020. P. 971–977. https://doi.org/10.1109/IV47402.2020.9304721

14. Mig-Fang Chang, John Lambert, Patsorn Sangkloy, et al. *Argoverse: 3D Tracking and Forecasting with Rich Maps*. In: Proc. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA: IEEE; 2019. P. 8748–8757. https://doi.org/10.1109/CVPR.2019.00895

15. Casas S, Gulino C, Suo S, et al. *The Importance of Prior Knowledge in Precise Multimodal Prediction*. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, NV: IEEE; 2020. P. 2295–2302. https://doi.org/10.48550/arXiv.2006.02636

Поступила в редакцию 09.04.2023
Поступила после рецензирования 25.04.2023
Принята к публикации 28.05.2023

Об авторах:

Сергей Александрович Иванов, старший инженер центра беспилотных технологий университета Иннополис (420500, РФ, г. Иннополис, ул. Университетская, 1), ORCID, se.ivanov@innopolis.ru

Бадер Рашид, руководитель отдела разработки систем распознавания центра беспилотных технологий университета Иннополис (420500, РФ, г. Иннополис, ул. Университетская, 1), ResearcherID, ScopusID, ORCID, b.rasheed@innopolis.university

Заявленный вклад соавторов:

С.А. Иванов — формирование основной концепции, цели и задачи исследования, проведение расчетов, подготовка текста, формирование выводов.

Б. Рашид — научное руководство, анализ результатов исследований, выдвижение возможных гипотез, доработка текста.

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Все авторы прочитали и одобрили окончательный вариант рукописи.

Received 09.04.2023
Revised 25.04.2023
Accepted 28.05.2023

About the Authors:

Sergey A. Ivanov, Senior Engineer, Center for Autonomous Technologies, Innopolis University (1, Universitetskaya St., Innopolis, 420500, RF), ORCID, se.ivanov@innopolis.ru

Bader Rasheed, Head of the Recognition Systems Development Department, Center for Autonomous Technologies, Innopolis University (1, Universitetskaya St., Innopolis, 420500, RF), ResearcherID, ScopusID, ORCID, b.rasheed@innopolis.university

Claimed contributorship:

SA Ivanov: basic concept formulation; research objective and tasks; computational analysis; text preparation; formulation of conclusions.

B Rasheed: academic advising; analysis of the research results; hypotheses advancement; revision of the text.

Conflict of interest statement: the authors do not have any conflict of interest.

All authors have read and approved the final manuscript.